Beliefs and Level-k Reasoning in Traffic

Eugene Vinitsky
UC Berkeley

Angelos Filos
University of Oxford

Nathan Lichtlé
ÃĽcole Normale SupÃľrieure

Kevin Lin
UC Berkeley

Nicholas Liu
UC Berkeley

Anca Dragan
UC Berkeley

Alexandre Bayen
UC Berkeley

Rowan McAllister
UC Berkeley

Jakob Foerster
FAIR

ABSTRACT

We consider the problem of sequential decision-making in partially observed strategic settings, such as driving with occlusions, where safety critical information may be hidden from any given agent. We set aside questions of perception and focus on the challenges of safe operation of a vehicle in the presence of these unobserved variables. In this paper, we bring in tools from the multi-agent literature and employ level-k reasoning – a hierarchical behavioural model in which each level is progressively more strategic. We outline common knowledge assumptions under which level-k reasoning can be used in driving settings without leading to infinite recursions and provide a practical implementation of our approach in several simple, demonstrative scenarios. In particular, Lk agents perform Bayesian filtering on unobserved variables by modeling other agents in the scene as L(k-1) agents and combine explicit inference with stochastic planning to generate an approximate best response. We empirically demonstrate that (a) L1 agents develop social perception, i.e., they infer hidden state variables by modeling others and performing counterfactual belief updates, and (b) given pro-social objectives L2 agents choose information revealing actions – a form of implicit communication.

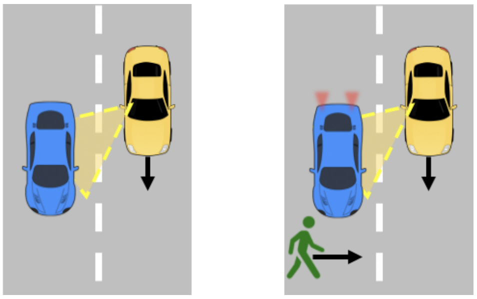
KEYWORDS

Autonomous Driving, Multi-Agent Systems, Common Knowledge, Theory of Mind, Simulation Theory, Model-Based Planning

ACM Reference Format:
Eugene Vinitsky, Angelos Filos, Nathan Lichtlé, Kevin Lin, Nicholas Liu, Anca Dragan, Alexandre Bayen, Rowan McAllister, and Jakob Foerster. 2021. Beliefs and Level-k Reasoning in Traffic. In *Manuscript submitted to Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), London, UK, May 3–7, 2021*, IFAAMAS, 9 pages.

1 INTRODUCTION

Real-world situations, such as driving, require agents to act in partially-observable, strategic environments. In the context of driving these situations are safety critical: a failure to catch occluded vehicles and pedestrians is estimated to be a significant fraction of fatalities [27]. Unfortunately, without the installation of intersection safety infrastructure that removes the occlusions, there is little that can be done directly about occlusions short of driving extremely slowly and cautiously.



(a) No Pedestrian (b) Occlusion

Figure 1: A yellow vehicle cannot see the illegally crossing pedestrian, since the blue vehicle obscures yellow's view of the pedestrian (depicted by yellow shading). (a) The blue vehicle continues to drive, which signals to the yellow vehicle that no pedestrian is hidden; (b) The decelerating blue vehicle enables the yellow vehicle to infer the presence of a hidden pedestrian and hence stop before getting involved into an accident.

Fortunately, there are usually multiple other agents in the scene. These agents implicitly communicate what is happening in these occluded regions through their actions. By taking the union of the visible regions, total coverage of the obscured scene is possible. If we maintain an explicit mental model of the behavior of other agents i.e., theory of mind [ToM; 5, 11, 23, 31], we can begin to reason about obscurities and reduce risk. This type of explicit reasoning has been demonstrated to be crucial for success [2] in tasks with hidden states, especially in non zero-sum settings [18].

For example, if we know what is visible to another vehicle, we can infer the absence of a pedestrian from a decision to accelerate or the presence of a pedestrian from an anomalously long stop. Figure 1b, depicts a scenario in which a pedestrian is obscured from the oncoming (yellow) vehicle. Importantly, this is a situation where short of building new infrastructure such as overhead cameras, there is simply no way for the AV to otherwise observe these pedestrians. However, by reasoning about why the blue car is starting to brake, the AV can infer the likelihood of pedestrians in occluded regions for better-informed decisions, e.g., slowing down and avoiding the accident. This inference is critical for safe AV navigation in partially observed road scenes.

Manuscript submitted to Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, London, UK. © 2021 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

Furthermore, once we have agents in the scene that are known to be inferring information about the world, we can construct pro-social agents that improve safety by bluffing. By moving appropriately, pro-social agents can trick other agents into believing that a pedestrian is present behind an obscurity. Fig. 2 depicts a scenario where this is key: the red vehicle could safely traverse without colliding with the green pedestrian, but, if it brakes, the yellow car will realize there is a pedestrian there and safely stop.

We seek to construct a framework which unifies both of these ideas: inference over observed actors and bluffing. We focus on partially-observed states due to occlusions and maintain an explicit belief over latent states [13]. Bayesian filtering [8] is used for belief inference and is made tractable using both a world model and a model of other agents. For modelling others, we apply tools from the multi-agent literature and employ level-k reasoning [LkR; 28] in partially-observable settings, demonstrating its usefulness in common traffic scenarios. In particular, level-k agents, i.e., Lk, perform Bayesian filtering on unobserved variables by modeling other agents in the scene as L(k-1) agents and performing explicit inference over their actions. Then, they use stochastic online planning to compute approximate best responses over the inferred beliefs.

We build a driving policy that operates safely around observed pedestrians and call this L0; it forms the base of the hierarchy we construct. In turn, L1's belief inference relies on treating every other agent in the scene as an L0 agent and interpreting their actions accordingly. Moreover, L1 agents give rise to the possibility of L2 agents, which can select actions that provide appropriate information to the inferring L1 agent. These L2 agents are shown to choose pro-social information revealing actions – a form of implicit communication. As we later discuss, this is a behavior autonomous vehicles could adopt to manipulate human scene perception and nudge them away from risky states. Furthermore, higher Lk levels may have additional safety inducing effects beyond those discussed here.

The main contribution of our paper is the introduction of concepts from the multi-agent literature to traffic and their application to a variety of risky driving scenarios. Under certain common knowledge assumptions, we demonstrate how the application of the level-k reasoning behavioural model to partially observed settings enables tractable Bayesian filtering for the latent state. Most importantly, we present driving scenarios where social perception and implicit communication lead to safer and improved driving experience, both unified under the Bayes-LkR framework.

2 PROBLEM SETTING AND NOTATION

We consider sequential decision-making in safety-critical partially occluded multi-agent driving scenes. First, we motivate and formally present our assumptions that make the problem tractable.

We represent our environments as finite-horizon general-sum partially-observed stochastic games [POSG; 20, 26]. Per usual notation, $n \in \{1, \ldots, N\}$ is the agent index and $t \in \mathbb{N}$ the discrete time index. The state space consist of per-agent factorised features $s_t = \cup_n s_t^n$. The transition dynamics satisfy the Markov property [24] with kernel $P_{\mathcal{G}}$, such that $s_{t+1}^n \sim P_{\mathcal{G}}(\cdot|s_t^n, a_t^n)$, where a_t^n is the n-agent's action.

Solving POSGs in their most general form is challenging, since agents cannot ground their knowns and their unknowns with others': (a) they do not know how others *perceive* the world and understand its dynamics; (b) they cannot expect how others *behave* and *reason* about certain situations and (c) they cannot escape the infinite regress of reasoning about others reasoning about their own reasoning ad infinitum.

To that end, we introduce *common knowledge* [CK; 9, 21] assumptions, enabling us to exit the infinite reasoning recursion and perform inferences about other agents' actions. Here, CK are things that all agents know that all agents know ad infinitum.

Assumption 1 (Common Knowledge World Model). *We assume access to a reliable world model, $P_{\mathcal{M}}(s_{t+1}|s_t, a_t^1, \ldots, a_t^N)$, which can be used for online planning (i.e. $P_{\mathcal{M}} \approx P_{\mathcal{G}}$) and is common knowledge to all agents.*

This constitutes assuming that all agents in the scene understand vehicle dynamics and have a shared model of pedestrian motion. This is a fairly strong assumption as human drivers might maintain quite different beliefs about pedestrian movement patterns.

For simplicity we also assume that the observation function for all agents consists only of a filtering operation that implements a restricted field of view but no aliasing: $o^n = w(s^n, s)$, where w reveals the list of all features visible to agent n given the arrangements of all other agents. Note that this accounts for occlusions amongst agents, i.e., when agent n blocks the view of agent j of the features of agent s^n.

Assumption 2 (Common Knowledge Observation Function). *We assume that the observation function, w, is common knowledge to all agents in the scene. Agent n does not have access to agent j's observation but it knows what agent j can or cannot observe.*

Note that this is a weak assumption that all agents understand how other drivers' vision works; this assumption may break if autonomous vehicles with varied sensing capabilities are introduced into the scene or if human drivers have an direction of focus controlling their field of view.

Next, we assume that agents agree on a base (a.k.a. *blueprint*) reactive policy, π_b, such that $a^n \sim \pi_b(\cdot|o^n)$. This corresponds to a policy that does not reason about others' actions but acts *conservatively* in the face of partial observability. For example, if our blueprint policy sees a tree that a pedestrian could be hiding behind and assesses that the pedestrian could collide with it, it will drive slowly and cautiously to give it sufficient time to brake and avoid a collision. Such a blueprint driving policy, can be either (a) provided [16], (b) learned via imitation learning [30] or reinforcement learning [15]. Formally,

Assumption 3 (Common Knowledge Blueprint Policy). *We assume access to a common knowledge blueprint (CK BP) reactive policy, π_b.*

Finally, to avoid issues of infinite recursion we need to ensure that two agents do not simultaneously run inference over each other. We refer to this as being "acyclic".

Assumption 4 (Clarity Of Levels). *We assume that, in each studied scenario, the obscurities are arranged such that there is no ambiguity in which agents might perform inference.*

This is a relatively strong assumption and frameworks for robust role assignment are left for future work. These assumptions enable a rich framework for sequential decision-making in safety-critical, partially-observable (occluded) multi-agent driving while still being grounded to the real-world setting, which we present next.

3 APPROACH

In this section, we describe a framework for online inference and planning algorithm for POSGs, while making use of tools from the multi-agent literature. We term our framework Bayes-L*k*R. Bayes-L*k*R combines Bayesian filtering [8] of the unobserved variables and planning using the inferred beliefs of these variables for the calculation of approximate best responses. Bayesian filtering is made tractable using the common knowledge world model (cf. Assumption 1) and L*k*R with the blueprint policy as the L0 agent (cf. Assumption 3). Model predictive control [7] is then used to find an approximate best response. We focus on settings that are acylic, where the proper role assignments are unambiguous and no agent is consequently required to model another agent of the same hierarchical level. This avoids possible recursions.

3.1 Bayesian Filtering for POSGs

Decision-making under uncertainty (i.e., occluded state) is performed by maintaining an explicit belief over the unobserved variables [4]. The belief is updated when new evidence (i.e., ego observations and others' actions) is provided.

Formally, let the action-observation history at time-step t for the n-th agent be the sequence of others' actions and ego actions, observations up to time-step t, given by

$$h^n_t = (o^n_1, a^{-n}_1, a^n_1, \dots, o^n_t, a^{-n}_t, a^n_t)\,. \tag{1}$$

We define the belief of the state of the n-th agent at time-step t as

$$b^n_t \triangleq p(s_t|h^n_t)\,. \tag{2}$$

Bayesian filtering [8] of the belief is given by the recursive definition

$$b^n_t \propto \underbrace{p(a^{-n}_t|s_t)}_{\text{“others” model}} \mathop{\mathbb{E}}_{s_{t-1}\sim b^n_{t-1}} \underbrace{P_{\mathcal{G}}(s_t|s_{t-1}, a^{-n}_{t-1}, a^n_{t-1})}_{\text{“dynamics” model}}, \tag{3}$$

where the proportionality holds by application of Bayes' rule and application of the Markov property twice.

We use the common knowledge world model, $P_{\mathcal{M}}$, (cf. Assumption 1) to approximate the "dynamics" model term in Eqn. (3) unless it is stated otherwise

$$\underbrace{P_{\mathcal{G}}(s_t|s_{t-1}, a^{-n}_{t-1}, a^n_{t-1})}_{\text{“dynamics” model}} \overset{\text{(CK)}}{\approx} \underbrace{P_{\mathcal{M}}(s_t|s_{t-1}, a^{-n}_{t-1}, a^n_{t-1})}_{\text{world model}}\,. \tag{4}$$

The expectation in Eqn. (3) is calculated exactly if it is tractable (e.g., the belief is a binomial distribution) or via Monte Carlo sampling otherwise. The model of the others is the joint policy of the observed agents, i.e.,

$$\underbrace{p(a^{-n}_t|s_t)}_{\text{“others” model}} \triangleq \pi^{-n}(a^{-n}_t|s_t)\,. \tag{5}$$

Next, we devise a recursive algorithm for "others" models based on the level-k reasoning (L*k*R) framework.

3.2 Approximate Level-k Reasoning for POSGs

The n-th self-interested agent in a POSG aims to maximise their expected cumulative reward, i.e., *returns*,

$$\max_{\pi^n} \mathop{\mathbb{E}}_{\tau\sim(b^n, P_{\mathcal{G}}, \pi^n, \pi^{-n})} \left[\sum_t r^n(s_t, s_{t+1}, a^n_t, a^{-n}_t)\right], \tag{6}$$

where the expectation is with respect to the *belief distribution*, environment stochasticity, ego and others' policies. When a world model (cf. Assumption 1) and a model of the "others" is available, the objective in Eqn. (6) can be solved via online stochastic planning, e.g., model predictive control [MPC; 7].

The framework of level-k reasoning (L*k*R) is a hierarchical behavioural model in which each level is progressively more strategic. L*k*R provides us with a recipe for building "others" models, used for both the Bayesian filtering (cf. Eqn. (3)) and the estimation of the returns (cf. Eqn. (6)). Level-k (Lk) agents model others as L(k-1) agents and therefore can be defined through the recursive definition

$$\pi_{\mathrm{L}k} \triangleq \arg\max_{\pi^n} \mathop{\mathbb{E}}_{h^n\sim\left(b^n, P_{\mathcal{G}}, \pi^n, \pi^{-n}_{\mathrm{L}(k\text{-}1)}\right)} \left[\sum_t r^n(s_t, s_{t+1}, a^n_t, a^{-n}_t)\right]. \tag{7}$$

In practice, the maximisation in Eqn. (7) cannot be performed exactly but only approximately, since (a) the expectations are approximated with Monte Carlo samples (b) only access to a world model (cf. Assumption 1) is provided and (c) global optimizers scale poorly. As a consequence, Lk agents are approximate best responses to L(k-1) agents. Our main contribution is the extension of the standard L*k*R framework to POSGs, reflected by the explicit belief updates with Bayesian filtering (cf. Eqn. 3) and marginalisation over the belief distribution (cf. Eqn. (7)). We term this framework Bayes-L*k*R.

By construction, L0 agents *react* to their private observation o^n: L0 agents react to their observations but they do not explicitly model others. L0 agents act according to the CK BP policy, π_b (cf. Assumption 3).

At the next level, L1 agents treat all other agents as L0 agents, i.e., $\pi^{-n} = \prod_{i\neq n} \pi^i_{\mathrm{L}0}$, while L2 agents treat others as L1 agents etc. Note that L2 agents maintain a belief over latent states as well as a belief over the belief of other L1 agents.

Next, we motivate the use of Bayes-L*k*R on strategically challenging driving scenarios.

3.3 Bayes-L*k*R in Traffic

We present driving scenarios where our Bayes-L*k*R demonstrate interesting behaviour, including social perception (L1 agents) and implicit communication (L2 agents). Future work lies in finding practical scenarios where higher-orders of reasoning, i.e., Lk for $k > 2$, are useful.

3.3.1 *L1 Agents: Social Perception.*

Consider the yellow car in Figure 1 to be an L1 agent. In both cases, the yellow car's field of view is blocked by the blue car and hence it cannot directly see if a pedestrian is at the crosswalk. Ignoring such a possibility and maintaining its speed may lead to a collision with a pedestrian, while over-conservative behaviour would be to always stop until either a pedestrian crosses or the yellow car finally manages to see behind the blue car. The L1 agent takes a different approach: It begins with representing its uncertainty about the presence (or absence) of a

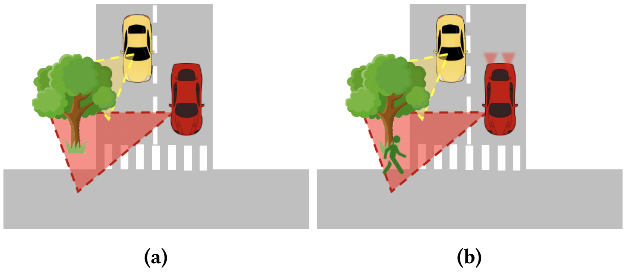


Figure 2: A yellow vehicle approaches an intersection, but cannot see any pedestrians, since the tree obscures yellows' view of the crosswalk (depicted by yellow shading). (a) The red vehicle crosses the intersection, which signals to the yellow vehicle that no pedestrian is hidden; (b) The decelerating red vehicle enables the yellow vehicle to infer the presence of a hidden pedestrian and hence stop before getting involved into an accident.

pedestrian with a belief over possible states of the world, encoded as a Bernoulli distribution, b^n_{t-1}. As the yellow L1 car agent observes the blue agent decelerating (accelerating) in Figure 1b (Figure 1a) it uses the CK BP policy, π_b, to justify the blue car's action, a^{-n}_t (cf. Eqn. 3). Then, it realises that the blue car would only decelerate (accelerate) if a pedestrian was (wasn't) at the crosswalk. Using this realization, the L1 agent (yellow car) updates its belief, inferring that a pedestrian is present (absent) from the scene and decides to stop (cross). Importantly, the L1 agent (yellow car) knows that the blue car can see if a pedestrian is at the crosswalk, or not, (cf. Assumption 2) and, as a result, even if it cannot directly observe what is happening at the crosswalk, it can distill useful information from the blue car's actions that reveal the state of the crosswalk.

This is a form of social perception, as defined by Sun et al. [29] – L1 agents treat other vehicles as distributed sensors that provide via their behaviors additional information about the environment beyond the scope of physical sensors. Note that this behaviour would naturally emerge from the Bayes-L*k*R framework as a best response.

3.3.2 L2 Agents: Implicit Communication. At one level higher of reasoning, L2 agents recognise that they are being modelled by L1 agents. L1 agents analyse the L2 agents' actions as if the L2 agents were L0 agents. L2 agents are *by construction* prosocial: they appreciate that they are modelled and act so that they *communicate* the most information about the occluded variables. Their actions are a form of implicit messages sent to L1 agents that take the role of receivers; the L1 agents model the L2 cars as L0 vehicles and can be manipulated into updated their beliefs via L2 actions.

A practical driving scenario where L2 agents are useful is depicted in Figure 2. In this scene, the red L2 car fully observes the state of the world, while the speeding yellow L1 car cannot see behind the tree where a pedestrian may be present (absent) as in Figure 2b (Figure 2a). The red car is ahead and can cross regardless of the presence or absence of a pedestrian. However, the L2 red car knows that it is being modeled by the yellow L1 car and decides to communicate the presence (absence) of the pedestrian by decelerating (accelerating) in Figure 2b (Figure 2a). This is a prosocial behaviour since the red car has to stop for the pedestrian even if their paths would not intersect in order to *signal* to the yellow car the presence of a pedestrian and avoid a potential accident.

4 CASE STUDIES

Here we present simple experiments illustrating that with a blueprint L0 model, inference can be used to reduce the crash risk when AVs appropriately adopt L1 and L2 policies. We discuss how we construct each level of the hierarchy and then present four potential scenarios where higher order reasoning is important.

Design of the L0 policy: To perform inference on our blueprint L0 policies, it is necessary that our blueprint policies be responsive to pedestrians so that inference is actually possible. This is a reasonable behavior as humans generally try not to hit pedestrians. If a vehicle approaches a crosswalk and observes a pedestrian crossing, it will begin to brake and aim to come to a full-stop before it reaches the crosswalk. In this case, let v be the speed of the vehicle, $a_{\text{min}} < 0$ its maximal deceleration and D its distance to the crosswalk. If it brakes with a constant deceleration $a < 0$, the vehicle will come to a stop after having traveled $d_{\text{stop}}(a) = \frac{v^2}{2|a|}$ meters. In our case, we consider two discrete braking behaviors: a soft braking with an acceleration of $a_{\text{soft}} = a_{\text{min}}/2$ and a hard braking with an acceleration of $a_{\text{hard}} = a_{\text{min}}$. We want the vehicle to stop within a safe margin d_{margin} of the crosswalk if there is a pedestrian crossing, while soft braking instead of hard braking when possible. To that end, the vehicle starts soft braking as soon as $d_{\text{stop}}(a_{\text{soft}}) + d_{\text{margin}} \geq D$ and until it stops. However, if that safe margin cannot be met with soft braking, the vehicle still soft brakes and starts hard braking as soon as $d_{\text{stop}}(a_{\text{hard}}) + d_{\text{margin}} \geq D$. Not that in our code implementation vehicles do not slow down or come to a stop at crosswalks in the absence of a pedestrian; this is a simplification to make implementation of the scenarios in the simulator easier but this restriction can be removed without changing any of the results.

In the absence of pedestrians, the vehicles obey the Intelligent Driver Model (IDM) [16], a ubiquitous model of human driving, to determine their accelerations. We assume that all vehicles appropriately use their turn signals so that their goals are known. This allows us to avoid collisions with other vehicles, by forecasting their trajectories 3 seconds into the future and hard braking if a collision is observed. Note that this type of simple blueprint policy would not be able to handle complex intersection scenarios and is primarily used as an illustrative tool for how interesting higher order policies can emerge even from a simple blueprint policy.

Design of the L1 policy: For the L1 controllers, we supplement the L0 policy with a rule wherein if the inferred pedestrian probability for the relevant crosswalk exceeds 80%, the L1 controller will assume there is a pedestrian in that position. This is a simple heuristic and could be replaced with something close to a best response using Model Predictive Control and sampling over the beliefs. However, we found this heuristic was sufficient for our demonstrations.

The inference procedure has two main steps:

- Look at the action of other cars in the scenario and then use the Gaussian model of the L0 vehicle to infer the probability of pedestrians given the action
- A linear dynamics update step where we evolve the probability of pedestrians appearing or disappearing.

This latter step uses a stochastic update matrix to update the probabilities. This matrix can be acquired from data or tuned to increase/decrease conservative behavior as the off-diagonal elements will increase/decrease the believed likelihood of pedestrians appearing or disappearing. Denoting the presence of a pedestrian as ped = 1 the dynamics evolve as

$$p\begin{pmatrix}\text{ped}=0\\ \text{ped}=1\end{pmatrix} = \begin{pmatrix}\text{ped}=0|ped=0 & \text{ped}=0|ped=1\\ \text{ped}=1|ped=0 & \text{ped}=1|ped=1\end{pmatrix}\begin{pmatrix}\text{ped}=0\\ \text{ped}=1\end{pmatrix}$$

In our case we simply set the diagonal elements to .99 and the off-diagonal elements to .01.

We outline the filtering procedure in Algorithm 1. Note that we only describe performing inference over a single agent as the multi-agent case can easily be extended from this procedure. A few notational elements: the crosswalks at an intersection are numbered 0 through 3 going clockwise from East and we refer to our belief about a pedestrian at crosswalk i as b_i. We use o_p^i as a binary indicator of a pedestrian at crosswalk i and $f_\pi(a|s_t, o_p^i)$ as the likelihood of the action under policy π.

Algorithm 1: Belief Maintenance of Agents in Occluded Regions using Bayesian Filtering

```
Input current belief $b_i \; \forall i \in 0, 1, 2, 3$
Input observation $o_t$
Input action of observed vehicle $a$
Compute joint-action likelihood w.r.t. counterfactual
  pedestrian observations $f_\pi(a|s_t, o_p^i) \; \forall i$
Compute normalizing factor $C = \sum_i f_\pi(a|o_t, o_p^i) b_i$
Compute the posterior probabilities $\frac{f(a|o_p^i) b_i}{C}$
Perform the dynamics update:
  $\begin{bmatrix} p(o_i = 0|a) \\ p(o_i = 1|a) \end{bmatrix} = F * \begin{bmatrix} p(o_i = 0|a) \\ p(o_i = 1|a) \end{bmatrix} \; \forall i \in 0, 1, 2, 3$
Return new beliefs $b_i \; \forall i \in \{0, 1, 2, 3\}$
```

Design of L2 Agents: For the L2 agents we use model predictive control (MPC) in combination with a simulator to compute the actions. Since we assume that all agents in the scene are either L1 or L0, we can perform this procedure without generating an infinite recursion. One complexity is how to simulate the L1 cars since we do not have access to their internal priors; we resolve this by initializing every visible L1 car with equal probabilities of pedestrian and not pedestrian. This procedure has one potential error mode: the L1 car may already be in the process of inference when we first observe it and so may be more likely to accelerate or decelerate than we expect. In the scenarios we consider there are only two cars so this error mode is not in effect, we leave studying the effects of this choice in the presence of more agents to future work.

We summarize the process of running the L2 controller in Alg. 2. For notational simplicity, we will assume that there is only one other car in the scene and that it is visible at the start. Furthermore, we will assume the other car is an L1 car; the L0 case follows naturally. We will use $\boldsymbol{v}$ to indicate the action recommended by MPC, $\boldsymbol{v}_i$ to be its i-th element and the superscript L1, L2 to indicate the L1 and L2 cars respectively. The MPC has access to two potential actions: braking, denoted a_{min} and imitating the action an L0 policy would take in the absence of pedestrians, denoted as a_{IDM}. Finally, the observation mask for vehicle L1 will be written as $w(s_t^{L1}, s_t)$.

Algorithm 2: MPC for an L2 Controller

```
Input acceleration bounds $a_{\text{min}}$, $a_{\text{IDM}}$, MPC horizon $H$,
  action sets $Q = \{a_{\text{min}}, a_{\text{IDM}}\}^H$, start state $s_0$, pedestrian
  beliefs for L1 car $b_{\text{ped}}^0 = [0.5, 0.5, 0.5, 0.5]$
Input model of L1 controllers $\pi_1(w(s_t^{L1}, s_t), b_{\text{ped}}^t)$
Input model of dynamics $s_{t+1} \sim P_{\mathcal{G}}(s_t, a_t^{-k}, a_t^k)$
for t = 0, 1, ..., T do
  $r_{\text{total}} = 0$
  for $q \in Q$ do
    $r_{\text{counter}} = 0$
    for $a_{L2} \in q$ do
      $s_{t+1} \sim P_{\mathcal{G}}(s_t, \pi_1(s_t, b_{\text{ped}}^t), a_{L2})$
      $b_{\text{ped}}^{t+1}$ = Algorithm 1($p_{\text{ped}}^{t+1}, a_{L2}$)
      $r_t = \begin{cases} v_{L2} & \text{if no collision} \\ -100 & \text{if collision} \end{cases}$
      $r_{\text{counter}}$ += $r_t$
    end for
    if $r_{\text{counter}} > r_{\text{total}}$ then
      $r_{\text{total}} = r_{\text{counter}}$
      $\boldsymbol{v} = q$
    end if
  end for
  $s_{t+1} \sim P_{\mathcal{G}}(s_t, \pi_1(w(s_t^{L1}, s_t), b_{\text{ped}}^t), \boldsymbol{v}_0)$
  $p_{\text{ped}}^{t+1}$ = Algorithm 1($p_{\text{ped}}^{t+1}, \boldsymbol{v}_0$)
end for
```

4.1 Occlusion Handling

For our model of occlusions we use a simple geometric intersection method. Each of the pedestrians and cars is represented as a rectangle. The view of a given vehicle k is a 120 degree cone of height 50 emanating from its center along the direction of motion as we assume drivers are looking straight ahead. We identify all objects within this cone as a set of potential blockers. Then, to check if k can see another object l, we check if the line of sight between the center of k and the center of l passes through any of the blockers. If so, we assert the k cannot see l. More complex models of obscurity that include head tilt and operate on non-rectangular objects are possible but this simple model serves sufficiently for purposes of demonstration.

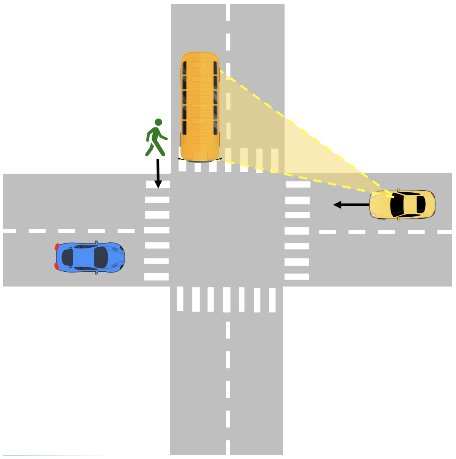


Figure 3: Scenario 1: an L1 car (yellow) approaches a crosswalk with a pedestrian obscured by a bus.

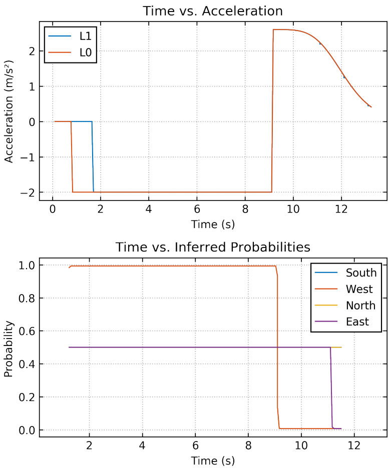


Figure 4: Predicted accelerations (top) and pedestrian probabilities (bottom) of the L1 and L0 vehicles for scenario 1. The L1 brakes shortly after the L0 does.

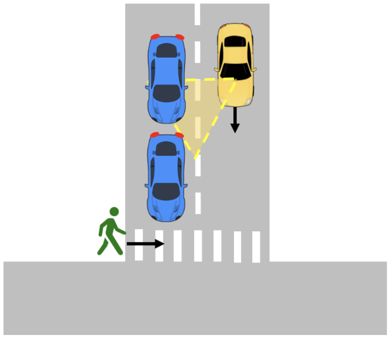


Figure 5: Scenario 2: an L1 car approaches a crosswalk with a line of cars blocking view of the pedestrian.

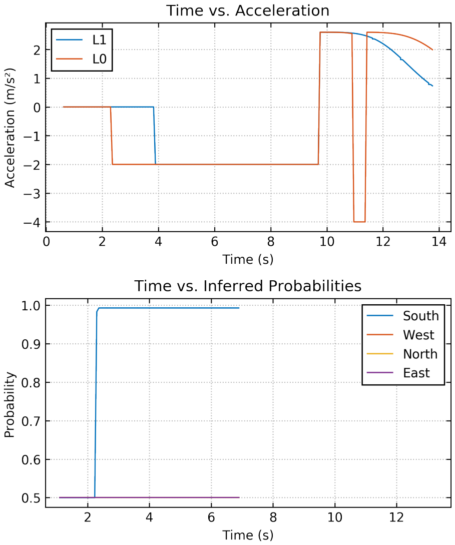


Figure 6: Predicted accelerations (top) and pedestrian probabilities (bottom) of the L1 and L0 vehicles for scenario 2. Only the L0 car closest to the crosswalk is plotted. The L1 car travels along the row of cars and brakes once it sees the car closest to the crosswalk. The predictions end once no other car is visible.

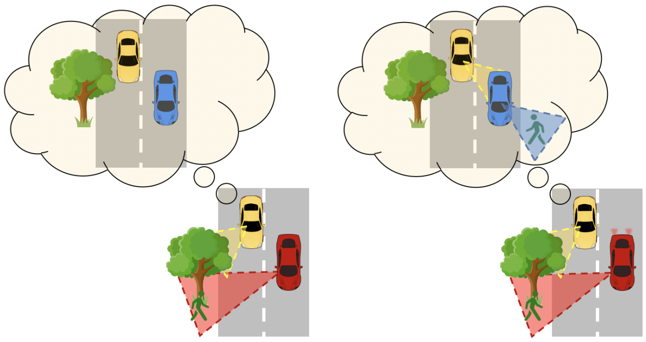
4.2 Scenarios

Here we present the scenarios that demonstrate compelling a need for explicit inference. The code to reconstruct these scenarios and all of our controllers is available at <ANONYMIZED>.

In the first scenario, shown in Fig. 3, a speeding L1 vehicle approaches an intersection where a pedestrian on the opposite crosswalk is obscured by a large bus. If the L1 vehicle waits until it sees the pedestrian to start braking, it will not have time to stop before it collides with the pedestrian. Fig. 4 illustrates the evolution of the inferred pedestrian probability: as the yellow L1 car gets close enough to see the blue L0 car, it quickly realizes that blue must be stopped because of a pedestrian either on the west or east edge. It waits till the pedestrian crosses at which point both vehicles go.

Fig. 5 presents a similar situation as Fig. 3 except here the pedestrian is obscured by a line of stopped cars. As the yellow L1 car travels along the line, it must infer that the stopped cars further from the crosswalk are stopped simply because they cannot travel any further while the car closest to the crosswalk is stopped because of a pedestrian. As seen in Fig. 6 it starts off along the line of cars and does not infer the pedestrian until it gets close to the front of the line, at which point it brakes. If the L1 car had not done that inference, it would not have had time to break and would have collided with the pedestrian.

Fig. 7 presents the L2 scenario where the pro-social L2 vehicle (red) is trailed by an L1 vehicle. The L2 vehicle spots a pedestrian ahead that L1 cannot see and must induce L1 to stop before it collides with it. This particular scenario is challenging challenging as the MPC horizon must be quite long to avoid entering states from which safety is infeasible. Fig. 8 demonstrates the interesting



(a) The L2 chooses not to break. Consequently the L1 vehicle, believing that the L2 vehicle is an L0 vehicle, (blue) assumes it is speeding up because no pedestrian is there.

(b) The L2 breaks. Consequently the L1 vehicle incorrectly assumes a pedestrian in front of the car it believes to be an L0 vehicle. Since it will later collide with the pedestrian, it slows down.

Figure 7: Scenario 3: An L2 vehicle (red) spots a pedestrian that it knows is hidden to the L1 vehicle behind (yellow) and that the speeding L1 vehicle will crash into if it continues.

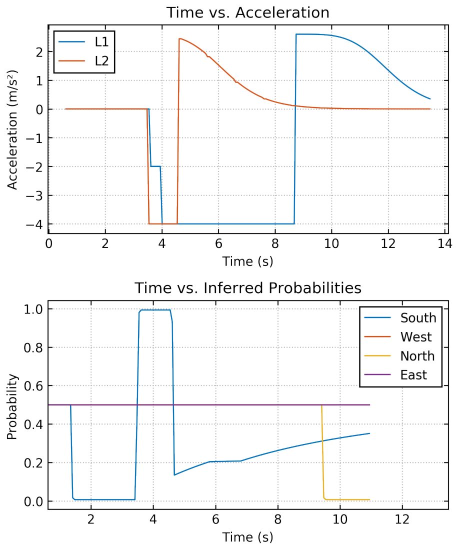
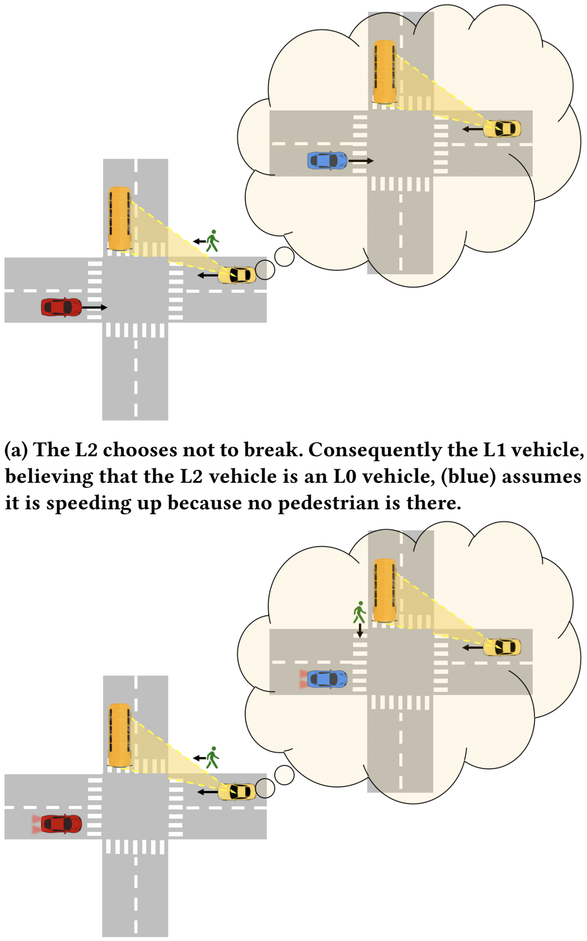


Figure 8: Accelerations (top) and predicted pedestrian probabilities (bottom) of the L1 and L2 vehicles for scenario 3. The L2 car hard brakes to trick the L1 car into thinking there is a pedestrian on the opposite crosswalk behind the bus. Note that the L1 continues to brake even after the inferred probability on the South crosswalk drops below 0.5 as the pedestrian has become visible.

behavior that emerges for the L2 controller: the MPC hard brakes, inducing a high belief in the presence of a pedestrian which leads to L1 hard braking as well. As soon as L1 can see the pedestrian on its own, L2 can accelerate again. Note that for this particular scenario with two lanes, we allow a more aggressive driving behavior where the car on the right lane is allowed to go over the crosswalk if the pedestrian is still on the left.



(a) The L2 chooses not to break. Consequently the L1 vehicle, believing that the L2 vehicle is an L0 vehicle, (blue) assumes it is speeding up because no pedestrian is there.

(b) The L2 breaks. Consequently the L1 vehicle incorrectly assumes a pedestrian in front of the car it believes to be an L0 vehicle and slows to avoid a future crash.

Figure 9: Scenario 4: An L2 vehicle (red) spots a pedestrian that it knows is invisible to the L1 vehicle behind (yellow) and that the speeding L1 vehicle will crash into if it continues.

Finally, Fig. 9 presents another example where L2 reasoning is useful. A distracted L1 (yellow) agent isn't paying attention and is unaware of a pedestrian to its right. The L2 car (red) realizes that if it breaks the L1 car will assume a pedestrian behind the bus and will begin to brake to avoid it. Fig. 10 shows this process in detail where the L2 car brakes for a few seconds, causing the predicted probability of a pedestrian on the west crosswalk to rise and inducing the L1 vehicle to stop.

5 RELATED WORK

The most closely related works on the line of finding schemes to handle unobserved variables in driving scenarios are [29] and [22]

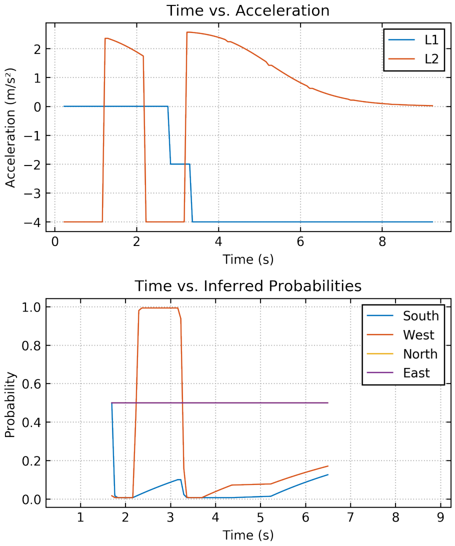


Figure 10: Accelerations (top) and predicted pedestrian probabilities (bottom) of the L1 and L2 vehicles for scenario 4. The L2 car hard brakes to trick the L1 car into being highly certain that there is a pedestrian on the West crosswalk. This leads to L1 decelerating before it sees the real pedestrian on the East crosswalk.

which respectively introduce L1 and L2 controllers, albeit implemented via different mechanisms than we use and without noting the connection to Level-k thinking. [29] introduces the notion of using other vehicles in the scene as distributed sensors, what we call L1 policies, for occluded states but use MPC and noisy rationality [32] to compute probabilities of observations. In contrast, we use Bayesian inference to acquire the probabilities using a policy rather than inference assuming noisy rationality. [22] introduces the L2 formulation as the outcome of the optimal policy in a *Bayesian Persuasian Game* [14] in which an ego vehicle acts to change another vehicle's belief about the state of the world. By assuming that belief probability updates are a Gaussian function of state, they derive a closed form cost function that includes the inference and that can be optimized to yield L2 actions. Our approach builds on these works and places them into one coherent framework. Our approach is also closely related to SPARTA [18] which uses common knowledge about the search procedures of other agents to avoid infinite recursion by using a random seed to allow each agent to perfectly reconstruct the search procedure of all other agents. Our particular common knowledge assumptions prevent us from having to use common knowledge assumptions over the search procedures of other agents but SPARTA could be used as an alternative.

There are also a variety of alternative approaches to handling obstructions at intersections. [17] focuses on having the autonomous car mimic groups of cars with low variance, assuming that generally mimicking humans is optimal. This procedure allows the AV to mimic behaviors like dodging a stalled car or giving an ambulance right of way; in many of our settings this practice would yield identical behavior as our L1 cars (braking because other agents braked). Instead of doing online inference as we do, offline belief space planning can be used to compute policies that are simply deployed as in [6] where they show that offline belief space planning methods can be used to scale to variable numbers of obscured vehicles. [3] use online belief space planning to drive an autonomous car in the presence of pedestrians with uncertain intentions.

There are also a variety of works that consider possible solutions to the problem of infinite regress (I model you modelling me modelling you etc.). [25] considers a similar concept as L2, namely AVs that consider that humans will perform a best response to the robot action but break the infinite regress by assuming that the robot acts first; this allows them to use a quasi-Newton method to pick the optimal action sequence for an MPC routine. We are also not the first to consider the application of level-k reasoning in driving settings, [19] uses a level-k model to handle conflicts at an intersection by iteratively computing best responses online; additionally they design an online procedure for assigning vehicles to roles. We extend the ideas from their work to scenarios with obscurity and clarify necessary common knowledge assumptions that make this possible.

While we focus on inference about the state of the world, there is also extensive work on inferring hidden states of drivers such as internal state or goals. For example, using the framework of noisy rationality, [1] looks at how consistent a given goal is with an optimal trajectory to that goal and use that to infer goal likelihood. The behavior we find emerges in L2 scenarios, taking actions to change another agent's belief's, is also studied as *legibility*, in which agents try to communicate hidden intents. [10] formulates a functional for computing legibility of trajectories, show that it can be optimized by gradient ascent, and construct a trust region method that keeps the resultant trajectories predictable. Instead of communicating desired goal states, [12] focuses on actions that correctly communicate the reward function. Many of these approaches are applicable to inferring world state and could be compared with our method.

6 CONCLUSIONS AND FUTURE WORK

In this work we formulated a mechanism by which autonomous vehicles can improve their safe operation in the presence of unobserved variables such as pedestrians. We define common knowledge settings under which the direction of influence of unobserved variables is clear and Level-k Reasoning can be applied without any recursion. We construct a variety of settings where inference combined with Level-k Reasoning can be used to improve the safety of driving in the presence of occluded pedestrians and show that at higher levels of reasoning pro-social behavior can be designed, pointing out a new mechanism by which autonomous vehicles can improve roadway safety. We show that these procedures can be tractably implemented using Bayesian inference and model predictive control and construct several case studies demonstrating the value of these approaches.

There are a few simplifying assumptions in this work whose relaxation would be an interesting assumption for future work. We assume access to a simplistic blueprint policy and our results are dependent on assumptions on how that blueprint policy operates; it would be interesting to construct a blueprint policy directly from human data and see if some of the scenarios we propose are feasible under more complex models of human driving. It is possible that given the variability in human driving it is not easy to perform accurate inference on unobserved variables.

For tractability of the Bayesian inference procedure, we assumed discrete pedestrian variables but realistic implementations could use a finer grid or continuous distributions. Finally, in the scenarios we constructed we assume that the assignment of vehicles to levels is known and fixed. Is it possible to construct schemes that relax this assumption, perhaps by performing inference over the roles? What happens if vehicles have incorrect assumptions about roles of the other vehicles; is there a procedure that might be robust to incorrect assumptions? These are all interesting directions for future work.

REFERENCES

[1] Stefano V Albrecht, Cillian Brewitt, John Wilhelm, Francisco Eiras, Mihai Dobre, and Subramanian Ramamoorthy. 2020. Integrating Planning and Interpretable Goal Recognition for Autonomous Driving. *arXiv preprint arXiv:2002.02277* (2020).

[2] Stefano V Albrecht and Peter Stone. 2018. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence* 258 (2018), 66–95.

[3] Haoyu Bai, Shaojun Cai, Nan Ye, David Hsu, and Wee Sun Lee. 2015. Intention-aware online POMDP planning for autonomous driving in a crowd. In *2015 ieee international conference on robotics and automation (icra)*. IEEE, 454–460.

[4] David Barber. 2012. *Bayesian reasoning and machine learning*. Cambridge University Press.

[5] Simon Baron-Cohen, Alan M Leslie, and Uta Frith. 1985. Does the autistic child have a “theory of mind”? *Cognition* 21, 1 (1985), 37–46.

[6] Maxime Bouton, Alireza Nakhaei, Kikuo Fujimura, and Mykel J Kochenderfer. 2018. Scalable decision making with sensor occlusions for autonomous driving. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2076–2081.

[7] Eduardo F Camacho and Carlos Bordons Alba. 2013. *Model predictive control*. Springer Science & Business Media.

[8] Zhe Chen et al. 2003. Bayesian filtering: From Kalman filters to particle filters, and beyond. *Statistics* 182, 1 (2003), 1–69.

[9] Christian Schroeder de Witt, Jakob Foerster, Gregory Farquhar, Philip Torr, Wendelin Boehmer, and Shimon Whiteson. 2019. Multi-agent common knowledge reinforcement learning. In *Advances in Neural Information Processing Systems*. 9927–9939.

[10] Anca Dragan and Siddhartha Srinivasa. 2013. Generating legible motion. (2013).

[11] Alison Gopnik and Henry M Wellman. 1992. Why the child’s theory of mind really is a theory. (1992).

[12] Dylan Hadfield-Menell, Stuart J Russell, Pieter Abbeel, and Anca Dragan. 2016. Cooperative inverse reinforcement learning. In *Advances in neural information processing systems*. 3909–3917.

[13] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence* 101, 1-2 (1998), 99–134.

[14] Emir Kamenica and Matthew Gentzkow. 2011. Bayesian persuasion. *American Economic Review* 101, 6 (2011), 2590–2615.

[15] Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. 2019. Learning to drive in a day. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 8248–8254.

[16] Arne Kesting, Martin Treiber, and Dirk Helbing. 2010. Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 368, 1928 (2010), 4585–4605.

[17] Nicholas C Landolfi and Anca D Dragan. 2018. Social Cohesion in Autonomous Driving. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 8118–8125.

[18] Adam Lerer, Hengyuan Hu, Jakob N Foerster, and Noam Brown. 2020. Improving Policies via Search in Cooperative Partially Observable Games.. In *AAAI*. 7187–7194.

[19] Nan Li, Ilya Kolmanovsky, Anouck Girard, and Yildiray Yildiz. 2018. Game theoretic modeling of vehicle interactions at unsignalized intersections and application to autonomous vehicle control. In *2018 Annual American Control Conference (ACC)*. IEEE, 3215–3220.

[20] Michael L Littman. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*. Elsevier, 157–163.

[21] Martin J Osborne and Ariel Rubinstein. 1994. *A course in game theory*. MIT press.

[22] Cheng Peng and Masayoshi Tomizuka. 2019. Bayesian persuasive driving. In *2019 American Control Conference (ACC)*. IEEE, 723–729.

[23] David Premack and Guy Woodruff. 1978. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences* 1, 4 (1978), 515–526.

[24] Martin L Puterman. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

[25] Dorsa Sadigh, Shankar Sastry, Sanjit A Seshia, and Anca D Dragan. 2016. Planning for autonomous cars that leverage effects on human actions.. In *Robotics: Science and Systems*, Vol. 2. Ann Arbor, MI, USA.

[26] Lloyd S Shapley. 1953. Stochastic games. *Proceedings of the national academy of sciences* 39, 10 (1953), 1095–1100.

[27] Akhil Shetty, Mengqiao Yu, Alex Kurzhanskiy, Offer Grembek, and Pravin Varaiya. 2020. Safety Challenges for Autonomous Vehicles in the Absence of Connectivity. *arXiv preprint arXiv:2006.03987* (2020).

[28] Dale O Stahl. 1993. Evolution of smartn players. *Games and Economic Behavior* 5, 4 (1993), 604–617.

[29] Liting Sun, Wei Zhan, Ching-Yao Chan, and Masayoshi Tomizuka. 2019. Behavior planning of autonomous cars with social perception. In *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 207–213.

[30] Bernard Widrow and Fred W Smith. 1964. Pattern-recognizing control systems.

[31] Heinz Wimmer and Josef Perner. 1983. Beliefs about beliefs: Representation and constraining function of wrong beliefs in young children’s understanding of deception. *Cognition* 13, 1 (1983), 103–128.

[32] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. 2008. Maximum entropy inverse reinforcement learning.. In *Aaai*, Vol. 8. Chicago, IL, USA, 1433–1438.